Towards Better Stability and Adaptability: Improve Online Self-Training for Model Adaptation in Semantic Segmentation

Dong Zhao, Shuang Wang
the School of Artificial Intelligence, Xidian University
zhaodong01@stu.xidian.edu.cn

Abstract

Unsupervised domain adaptation (UDA) in semantic segmentation transfers the knowledge of the source domain to the target one to improve the adaptability of the segmentation model in the target domain. The need to access labeled source data makes UDA unable to handle adaptation scenarios involving privacy, property rights protection, and confidentiality. In this paper, we focus on unsupervised model adaptation (UMA), also called source-free domain adaptation, which adapts a source-trained model to the target domain without accessing source data. We find that the online self-training method has the potential to be deployed in UMA, but the lack of source domain loss will greatly weaken the stability and adaptability of the method. We analyze the two possible reasons for the degradation of online self-training, i.e. inopportune updates of the teacher model and biased knowledge from source-trained model. Based on this, we propose a dynamic teacher update mechanism and a training-consistency based resampling strategy to improve the stability and adaptability of online self-training. On multiple model adaptation benchmarks, our method obtains new state-of-the-art performance, which is comparable or even better than state-of-the-art UDA methods.

1. Introduction

Unsupervised Domain Adaptation (UDA) has received extensive attention on semantic segmentation tasks [49, 59, 60, 63], which transfers the knowledge in the source domains (*e.g.* synthetic scene) to the target ones (*e.g.* real scene). UDA in semantic segmentation aims to alleviate the dependence of deep neural network-based models on dense annotations [18, 46, 61] and improve their generalization ability to target domains [5, 12, 15]. However, in proprietary, privacy, or profit-related concerns, source domain data is often unavailable, which presents new challenges for UDA [9, 27, 55]. To this end, the setting of Unsupervised Model Adaptation (UMA) is proposed [6, 9, 21, 30, 35], aiming to adapt the source-trained model to the unlabeled target domain without using source domain data.

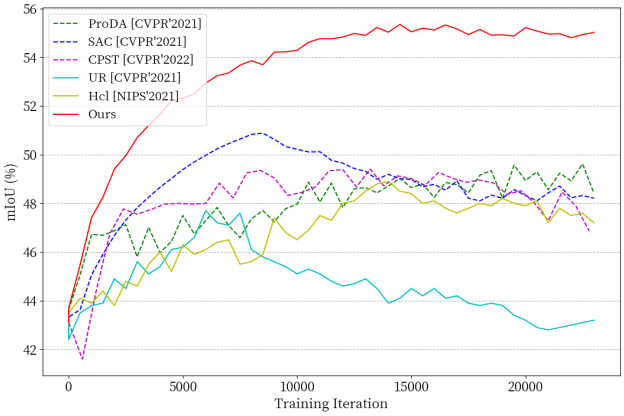

Figure 1. Under the unsupervised model adaptation (UMA) setting, the mIoU score (%) of different methods on the validation set throughout the training in GTA5 $\rightarrow$ Cityscapes adaptation task. The dashed line represents the self-training UDA methods, and the solid line represents the MDA methods.

In UMA, the knowledge in the source-trained model becomes the only available supervision signal, making self-training on pseudo-labels the mainstream in the field. Most existing UMA methods [26, 36, 57] adopt offline self-training methods, which iteratively updates the pseudo-labels and retrains the models. Although some improvements have been made, iterative self-training requires expert intervention [1, 59], as ill-suited rounds and termination often make it under-adapted.

The recently proposed online self-training (ONST) methods [1, 31, 59] in UDA avoid the iterative training by online co-evolving pseudo labels, showing great potential. Then can ONST be applied to UMA scenarios without accessing source data? We deploy the state-of-the-art ONST methods ProDA [1], SAC [59] and CPST [31] to UMA and draw the mIoU score curve on the validation set during training, as shown in Fig. 1. These ONST methods (dashed line) achieve more competitive performance than existing UMA methods (solid line). Nevertheless, taking a closer look at the curves in Fig. 1, these ONST methods

present different degrees of degradation and unstable adaptation process. Besides, their best performance in UMA decreased by $4\% - 5\%$ mIoU scores on average than in UDA (See Table 1 and 2). Consequently, we conclude that existing ONST methods suffer from impaired stability and adaptability when applied to UMA.

This paper is committed to improving the stability and adaptability of ONST methods in UMA. To begin with, we explore two reasons for the poor stability and adaptability of ONST in UMA. (1) The inopportune update of the teacher model causes the failure of co-evolution because the teacher model will continuously aggregate unevolved students. Concretely, as the teacher becomes the only supervisor in UMA, rapid updating will make the student lose the direction of evolution, and slow updating will make the student overfit the historical supervision, all of which leads to humble benefits of teachers' updating. (2) The bias towards minority categories in the source-trained model results in insufficient adaptation to those minorities as the bias is easily amplified in ONST, even with heuristic [1] or prototype thresholding [59] being set.

Next, we present the explored solutions. For (1), we find that the student's performance on historical samples during evolution can feedback on whether the student model has evolved. Consequently, we propose a Dynamic Teacher Update (DTU) mechanism. DTU explores two feedback signals by information entropy [13] and soft neighborhood density [45], which can better assess the evolutionary state of students. DTU then dynamically controls the update interval of the teacher model according to the students' feedback to aggregate more evolved students. For (2), we find that resampling minority categories can effectively alleviate the bias towards minorities in UMA. However, most existing resampling strategies [10, 11, 20, 50] rely on the source data and cannot apply in UMA. To this end, we propose a Training-Consistency based Resampling (TCR) strategy. TCR adaptively estimates the biased categories from the being-adapted model and selects reliable samples in biased categories as resampling candidates. Through these efforts, our method greatly improves the stability and adaptability of ONST in UMA, as shown in Fig. 1 (red solid line). We refer our method to *DT-ST*, as *D*TU and *T*CR play critical parts in online *S*elf-*T*raining under UMA.

Sufficient experiments show that the proposed *DT-ST* further exploits the potential of online self-training in UMA, towards better stability and adaptability. Moreover, *DT-ST* obtains new state-of-the-art performance on different UMA benchmarks and achieves comparable or even better performance than advanced UDA methods.

2. Related Work

Unsupervised Domain Adaptation (UDA) technology can often be summarized into the following three types for semantic segmentation tasks.

1) *Domain alignment* adopts adversarial training [4, 19, 22, 39, 52, 53] or statistics matching [23, 24, 38, 54, 62] to align distribution between the source and target domain at certain space, so that the classifier trained in the source domain can be applied to the target. Specifically, the imaging style at the input space [4, 17, 25, 33, 56, 61], the statistics at the feature space [39, 41, 52, 53], and the layout at the output space [19, 51, 52] are mainstream aligned objects.

2) *Offline self-training* utilizes the source-trained or domain-aligned model to generate pseudo labels, and then iteratively fine-tune the model and update the pseudo labels [28, 34, 40, 48, 63, 64]. Works along this line mine target traits by learning high-quality pseudo-labels. To this end, heuristic threshold strategies [63, 64] and reliable sample selection strategies [28, 29, 34, 40] are designed to generate high-quality pseudo labels, which can better enhance the adaptation performance.

3) *Online self-training* [1, 7, 14, 31, 59] adopt online co-evolving pseudo labels to avoid iterative training. Most of them adopt a teacher model to guide the evolution of a student model. Then the evolved student model is aggregated into the teacher to achieve co-evolution.

Unsupervised Model Adaptation(UMA) [9, 27, 55, 57] is proposed for model adaptation in confidential and privacy scenarios. In semantic segmentation task, most advanced UMA work [9, 26, 57] uses offline self-training to solve this problem, which will inevitably encounter the bottlenecks of expert intervention and parameter tuning that is not easy to deploy. [1, 59] Huang *et al.* [21] propose an online self-training method called HCL for model adaptation. To stable self-training, HCL constrains current models to focus on knowledge consistent with historical models. However, the over-regularization of historical models may lead to slow evolution. In contrast, our method can dynamically balance historical regular and pseudo label updates, and achieve better stability and adaptability.

Class Imbalance in UDA is a thorny problem [1, 20, 59, 63], due to the lack of attention and large semantic shift across domains. To solve this problem, threshold policy [1, 59, 63, 64] and resampling strategies [10, 11, 20, 50] are designed. The principle of threshold policy is to design a lower threshold for the minority category for picking out more pseudo-labels, as minority categories are often underperforming and show lower confidence probabilities. In particular, ProDA [59] designs a few-category-friendly prototype classifier to achieve the above goals without manual thresholding. However, we found that such methods are not enough to mitigate the bias of source-trained models due to insufficient incentives for minority classes. Most resampling strategies [10, 11, 20, 50] perform whole-image or local-patch paste-copy from source domain images containing minority categories. Although effective, the reliance on

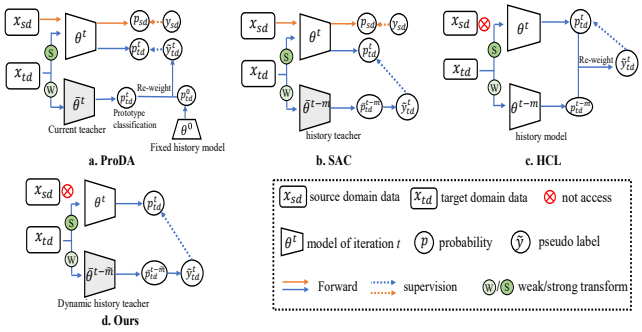

Figure 2. Schematics of different online self-training methods. The subfigure a, b and c come from [59], [1] and [21] in turn. In d, $\tilde{m}$ denotes a dynamic update interval instead of fixed one.

the source data limits its application to UMA scenarios. Our method achieves minority resampling only using target domain data, freeing resampling techniques from dependence on the source data.

3. Method

3.1. Problem Definition

In this part, we give the definition of the problem. Let $D_{sd} = \{(x_{sd}, y_{sd})\}$ be the labeled source domain data, $D_{td} = \{x_{td}^n\}_{n=1}^N$ be the unlabeled target domain data. The D_{sd} and D_{td} share K categories. Let G and θ be the source-trained segmentation model and its parameters. In Unsupervised Domain Adaptation (UDA), all these elements are accessible, the goal is to adapt the model G to D_{td}. In Unsupervised Model Adaptation (UMA), we keep the same goal but the source domain data is not accessible.

3.2. Online Self-training in UDA

We briefly introduce the online self-training (ONST) methods [1, 59] in UDA semantic segmentation in a generic form. ONST maintains a student model G_S and an online updated teacher model G_T. For the student model, a two-part loss is used for supervision. The first is the supervised loss of the source domain, *i.e.*, $L_{sd} = H[x_{sd}, y_{sd}]$, where H is the pixel-level cross-entropy loss. The second is the unsupervised weak-to-strong consistency loss of the target domain,

$$L_{td} = H[G_{stu}(\mathcal{W}(x_{td})), G_{tea}(\mathcal{S}(x_{td}))]. \quad (1)$$

$\mathcal{W}$ and $\mathcal{S}$ are the weak and strong image transformations, respectively. The weak-to-strong transformation enables the teacher to generate better pseudo-labels than the student. For the teacher model, its parameters are the student's momentum-updated version, *i.e.*, $\bar{\theta}_{t+1} = \gamma\theta_t + (1-\gamma)\bar{\theta}_t$. $\bar{\theta}_t$ and θ_t are the teachers' and students' parameters at the t-th iteration, respectively. γ is the update weights.

In the above paradigm, updating both teacher and student simultaneously will lead to severe model degradation due to the lack of a purposeful evolutionary direction. To this end, these ONST methods [1, 59] adopt historical supervision to regularize the direction of model evolution. For example, ProDA [59] and SAC [1] uses a fixed initial teacher or a slowly updated teacher as historical supervision, respectively. Their schematic can be seen in Fig. 2. For the convenience of discussion, we uniformly express ONST in the form of SAC, as the historical supervision in ProDA can be regarded as a special form of that in SAC. Then Eq.1 can be rewrite as follows,

$$L_{td} = H[G_{stu}^{\theta_t}(\mathcal{W}(x_{td})), G_{tea}^{\bar{\theta}_{t-m}}(\mathcal{S}(x_{td}))]. \quad (2)$$

G^{θ_t} denotes the model with parameters θ_t. m denotes the update interval of the teacher model. The larger it is, the slower the teacher update. Then, the update formula for the teacher model can be rewritten as follows,

$$\bar{\theta}_t = \gamma\theta_t + (1-\gamma)\bar{\theta}_{t-m}, \quad (3)$$

3.3. Online Self-training in UMA

In UMA, the source data is not accessible, which makes the student's supervision only from the weak-to-strong consistency loss, *i.e.*, Eq. 2. In that case, as mentioned before, ONST becomes unstable and prone to degradation. We present our analysis as follows.

Intuitively, an important condition for co-evolution is that the teacher model can continuously aggregate the evolved student by Eq. 3. Then, in each update of the teacher, whether the student evolves becomes the key factor for co-evolution. So, how to evaluate the evolutionary state of the student model in Eq. 3? We conjecture that changes in student model performance over historical samples during an update period can reflect evolution. To verify the guesses, we perform the following experiments. Let the sampling set in each teacher update interval be D_{td}^m. We evaluate the mIoU score of the student model before and after m optimizations (*i.e.*, $G_{stu}^{\theta_{t-m}}$ and $G_{stu}^{\theta_t}$) on D_{td}^m. If the mIoU score of $G_{stu}^{\theta_t}$ is higher than that of $G_{stu}^{\theta_{t-m}}$, it is recorded as a gain. Then, we accumulate the number of gains and calculate the average gain rate GR. Formally, GR is calculated as follows,

$$\mathrm{GR} = \frac{1}{T}\sum_{t}^{T}\delta(V[G_{stu}^{\theta_{t-m}}(D_{td}^m)], V[G_{stu}^{\theta_t}(D_{td}^m)]), \quad (4)$$

where $V[\cdot]$ is the mIoU evaluation function. $\delta(\cdot,\cdot)$ is a comparison function. If the former is large, $\delta(\cdot,\cdot)$ returns 0; Otherwise, it returns 1.

We set a series of m values and plot the curves of mIoU score on the validation set of the teacher model and GR

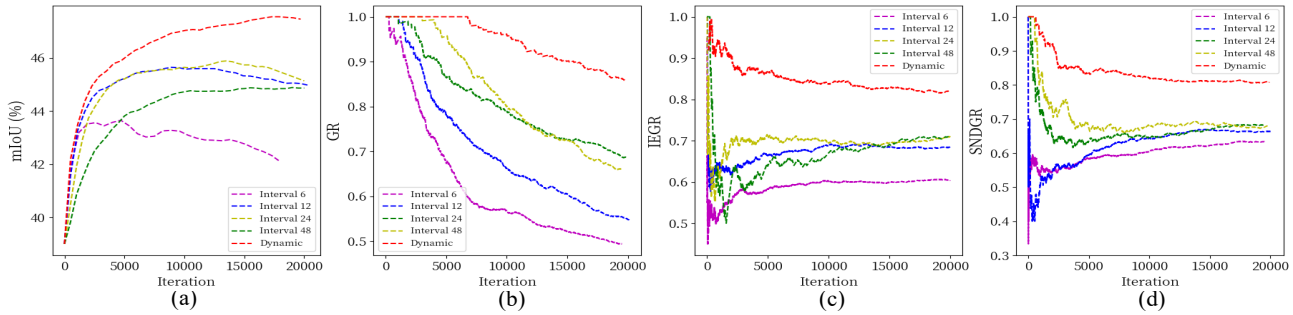

Figure 3. Training curves for different teacher update intervals.

with training, as shown in Fig. 3 (a) and (b). A comprehensive comparison of (a) and (b) can find the following conclusions: ① A more stable self-training tends to maintain a higher GR, meaning that co-evolution and GR are positively correlated. Thus, it's reasonable to argue that the performance of the student model over historical samples can approximately reflect its evolutionary state. More importantly, ② we argue that the inappropriate setting of the teacher's update interval m prevents the teacher from aggregating the evolved student, resulting in model degradation, *e.g.*, too fast (purple line) or too slow (green line) in Fig. 3 updates will not keep high GR.

3.4. Dynamic teacher update mechanism

Based on the above conclusions, we can monitor the student performance on historical data D_{td}^m, *e.g.*, set $V[G_{stu}^{\theta_{t-m}}(D_{td}^m)] > V[G_{stu}^{\theta_t}(D_{td}^m)]$ to control the updates interval, instead of setting a fixed interval m. In this way, the update interval can be be set dynamically and appropriately by the student's feedback, which will facilitate co-evolution. However, $V[\cdot,\cdot]$ is the mIoU function requiring the target domain labels. So next, we explore functions that can replace V without using target domain labels.

Information entropy [13] is often used as a measure of the uncertainty of model output. If the model output is low-entropy, it is considered reliable. To verify its feasibility, we replace the evaluation function $V[\cdot]$ in Eq. 4 with information entropy function $E[\cdot]$ and define average information entropy gain rate (IEGR) as follows,

$$\text{IEGR} = \frac{1}{T}\sum_{t}^{T}\delta(E[G_{stu}^{\theta_t}(D_{td}^m)], E[G_{stu}^{\theta_{t-m}}(D_{td}^m)]). \quad (5)$$

We draw the change curve of IEGR with training, as shown in Fig. 3 (c). It shows that the overall running trend of IEGR is close to that of EGR, but they differ in local trends and magnitudes of changes.

The soft neighborhood density (SND) [45] adopts the cluster density to determine whether the model is well adapted. The basis is that the feature structure of a well-adapted model should be compact. SND is defined as,

$$\text{SND} = E[\text{softmax}(p * p^T)]. \quad (6)$$

p is the output probability map by the model with shape $wh \cdot K$. Inspired by this, we verify the feasibility of SND as a feedback. We then replace the evaluation function $V[\cdot]$ in Eq. 4 with SND$[\cdot]$ and introduce average SND gain rate (SNDGR). The formula will not be repeated. Its change curve with training is shown in Fig. 3 (d). SNDGR shows a more similar trend to EGR, which shows that SND has potential as a feedback signal.

After that, we introduce the dynamic teacher update (DTU) mechanism to control the teacher update interval. The details are in Algorithm 1. The feedback function F in Algorithm 1 can be $1 - E[\cdot]$ (DTU-E) or SND$[\cdot]$(DTU-SND). Meta maximum evolution iteration $M$is a hyper-parameter, which prevents teachers from not updating for long time due to some extreme conditions. The red curves in Fig. 3 is using DTU-SND. Compared with fixed update mechanism, DTU shows better stability and adaptability.

Algorithm 1 Dynamic teacher update mechanism.

Input: Student model G_{stu}^{θ}, teacher model $G_{tea}^{\bar{\theta}}$, target domain data D_{td}, feedback function V and meta maximum evolutionary iteration M.
Output: Optimized teacher model G_{tea}
Copy G_{stu}^{θ} as G^{θ_0}; $U = 0$
for sample batch $\{x_{td}^b\}_{b=1}^B \in D_{td}$ **do**
  Optimizing Eq. 2 with D_{td}^b, $G_{tea}^{\bar{\theta}}$, and G_{stu}^{θ}
  Update G_{stu}^{θ}
  Store $\{x_{td}^b\}_{b=1}^B$ to D_{td}^B ; U++
  if $V[G^{\theta_0}(D_{td}^B)] < V[G^{\theta}(D_{td}^B)]$ **or** $U > M$ **then**
    Update $G_{tea}^{\bar{\theta}}$ by Eq. 3
    Copy G_{stu}^{θ} as G^{θ_0}; $U = 0$
return $G_{tea}^{\bar{\theta}}$

3.5. Training-consistency based Resampling

As an effective strategy to alleviate the imbalance problem, Copy-Paste resampling [10, 11, 50] is widely used in

the UDA field. It uses the class statistical distribution of the source domain to determine the minority categories, copies the source domain images containing minority categories, and pastes them to the target. However, the inaccessible source data prevents these methods from being applied to UMA. To this end, we propose a training-consistency based resampling (TCR) strategy for UMA. TCR adopts online estimated class distribution to determine minority categories, and exploits training consistency as the criteria for selecting copy objects. Specifically, we maintain an online average class score ACS from the output probability of the teacher model as follows,

$$\mathrm{ACS}_k^t = \frac{1}{hw}\sum_{i,j} p_{i,j,k}^t, \tag{7}$$

where p^t is the probability map with resolution $h \times w$ generated by the $G_{tea}^{\bar{\theta}_t}$. The ACS is updated by exponential moving average, *i.e.*, $\mathrm{ACS}_k^t = \alpha \mathrm{ACS}_k^{t-1} + (1-\alpha)\mathrm{ACS}_k^t$. The weight α is set as 0.999. The low-confidence categories in ACS are considered to be minorities, because they are always at a disadvantage in competition with other categories. We then use ACS to determine the sampling rate SR_k for k-th category as follows,

$$\mathrm{SR}_k = \mathrm{Normalize}(1 - \mathrm{ACS}_k), \tag{8}$$

where $\mathrm{Normalize}(x)_i = \frac{x_i}{\sum_j x_j}$. Through Eq. 8, the sampling rate of the minority category will be greatly improved.

Next, we build reliable target candidates for copy-paste operation. We regard the prediction consistency of the teacher model before and after multiple iterations of evolution as reliability. The motivation is that model evolution in self-training is reflected in the correction of uncertain regions. Thus, it is reasonable to argue that the prediction not affected by model evolution is highly reliable. Concretely, after each I iterations, for any sample $x_{td}^n \in D_{td}$, we calculate the corresponding reliability ReL^n as follows,

$$\mathrm{ReL}^n = \mathrm{IoU}[\xi(G_{tea}^{\bar{\theta}_{t-I}}(x_{td}^n)), \xi(G_{tea}^{\bar{\theta}_t}(x_{td}^n))], \tag{9}$$

ξ is a labeling function that converses soft predictions to hard labels. $\mathrm{IoU}[\cdot,\cdot]$ is the IoU evaluation function of K categories. ReL^n is a K-dimensional IoU score vector. The larger the element in ReL^n, the higher the reliability of the corresponding category. For each class k, we sort all target domain images according to ReL_k, and take the samples from the top C as the candidates for copy-paste.

4. Experiments

4.1. Experimental Setup

We perform Unsupervised Model Adaptation (UMA) using pre-trained models from two simulation datasets GTA5 [43] and SYNTHIA [44], adapting them to real scenarios including the single-domain Cityscapes dataset [8] and the mixed-domain BDD-100k dataset [58]. As in previous protocol [12,21,36], we uses the training set of both real-world datasets as the target domain training data and the validation set as the testing data. We evaluate the segmentation performance with per-class Intersection-over-Union (IoU) and the mean IoU (mIoU).

4.1.1 Datasets.

Synthetic datasets GTA5 dataset contains 24,999 virtual urban scene images with a resolution of 1914×1024. SYNTHIA dataset is rendered from a virtual city, which provides 9,400 images with a resolution of 1280×760. On GTA5 and Synthia dataset, we perform UMA with 19 and 16 common semantic categories, respectively.

Real-world datasets Cityscapes dataset provides 3,975 real urban scene images from 50 different cities in primarily Germany, with a resolution of 2048×1024. BDD-100K [58] is another real-world dataset collected from various locations in the US, which contains diverse scene images (*e.g.* rainy, snowy, and cloudy image) with a resolution of 1280×720.

4.1.2 Implementation Details.

We adopt the Deeplab-v2 [3] as the segmentation model with ResNet-101 [16] and VGG-16 [47] as the feature extractor and the aspp [3] module as the classifier. In data processing, all target domain images are resized to the same shorter edge while preserving aspect ratios and then input to the teacher model. The input to the teacher model is a weak augmented version, including random flipping and small range random scaling. The input to the student model is a corresponding strong augmented version, including Gaussian blur, colour jitter, and random center cropping. We follow the augmentation parameters in [59]. During training, we apply the SGD optimizer [2] with the momentum of 0.9. The initial learning rate is set to 2.5×10^{-4}, and then is reduced following a poly policy with a power of 0.9. The batch size is set as 4. We train our framework for 20,000 iterations on all UMA tasks, using an RTX3090 GPU (24GB). For parameter setting, the teacher update weight γ in Eq. 3 is set to 0.99. The meta maximum evolutionary iteration M in Algorithm 1 is set to 50. In the training-consistency based resampling strategy, the iteration number I for reliability evaluation is set to 4000 and the top 50% candidates are select.

4.2. Comparison to state of the art

We compare our method with previous state-of-the-art methods, including source-free unsupervised model adaptation (UMA) methods and unsupervised domain adaptation (UDA) methods. Our method significantly boosts the

	SF	road	sidewalk	Building	Wall	fence	pole	light	sign	vege.	terrain	sky	person	rider	car	truck	bus	train	mbike	bike	mIoU
IAST (ECCV'2020) [40]	✗	93.8	57.8	85.1	39.5	26.7	26.2	43.1	34.7	84.9	32.9	88.0	62.6	29.0	87.3	39.2	49.6	23.2	34.7	39.6	51.5
MetaCorr (CVPR'2021) [14]	✗	92.8	58.1	86.2	39.7	33.1	36.3	42.0	38.6	85.5	37.8	87.'6	62.8	31.7	84.8	35.7	50.3	2.0	36.8	48.0	52.1
ProDA (CVPR'2021) [59]	✗	91.5	52.4	82.9	42	35.7	40	44.4	43.3	87	43.8	79.5	66.5	31.4	86.7	41.1	52.5	0	45.4	53.8	53.7
SAC (CVPR'2021) [1]	✗	90.4	53.9	86.6	42.4	27.3	45.1	48.5	42.7	87.4	40.1	86.1	67.5	29.7	88.5	49.1	54.6	9.8	26.6	45.3	53.8
CPST(CVPR'2022) [31]	✗	91.7	52.9	83.6	43	32.3	43.7	51.3	42.8	85.4	37.6	81.1	69.5	30	88.1	44.1	59.9	24.9	47.2	48.4	55.7
Source model		65.0	16.1	68.7	18.6	16.8	21.3	31.4	11.2	83.0	22.0	78.0	54.4	33.8	73.9	12.7	30.7	13.7	28.1	19.7	36.8
URMDA (CVPR'2021) [9]	✓	92.3	55.2	81.6	30.8	18.8	37.1	17.7	12.1	84.2	35.9	83.8	57.7	24.1	81.7	27.5	44.3	6.9	24.1	40.4	45.1
SFDA (CVPR'2021) [36]	✓	91.7	52.7	82.2	28.7	20.3	36.5	30.6	23.6	81.7	35.6	84.8	59.5	22.6	83.4	29.6	32.4	11.8	23.8	39.6	45.8
SDF (MM'2021) [57]	✓	95.2	40.6	85.2	30.6	26.1	35.8	34.7	32.8	85.3	41.7	79.5	61	28.2	86.5	41.2	45.3	15.6	33.1	40.0	49.4
HCL (NIPS'2021) [21]	✓	92.0	55.0	80.4	33.5	24.6	37.1	35.1	28.8	83.0	37.6	82.3	59.4	27.6	83.6	32.3	36.6	14.1	28.7	43.0	48.1
DT-ST (Ours)	✓	90.3	47.8	84.3	38.8	22.7	32.4	41.8	41.2	85.8	42.5	87.8	62.6	37.0	82.5	25.8	32.0	29.8	48.0	56.9	52.1
Source model + DG [32]		80.2	30.2	79.6	30.7	20.3	31.9	36.1	18.6	80.6	23.9	75.2	63.0	36.2	84.8	31.2	36.1	4.4	31.2	28.0	43.3
ProDA†(CVPR'2021) [59]	✓	85.6	45.4	76.5	40.1	31.9	38.9	36.4	47.4	85.8	45.7	80.1	63.6	0	85.6	33.7	51.2	0	37.6	52.3	49.4
SAC†(CVPR'2021) [1]	✓	89.1	52.7	82.1	40.3	26.7	40.7	44.1	40.1	81.6	40.1	81.6	67.4	26.1	85.1	44.5	48.8	3.8	26.4	43.1	50.8
CPST†(CVPR'2022) [31]	✓	86.7	38.6	82.2	39.8	32.1	40.8	41.5	43.2	85.6	42.8	73.6	65.5	22.1	87.3	27.1	41.1	0	37.6	49.5	49.3
HCL (NIPS'2021) [21]	✓	92.6	54.6	82.8	33.2	26.2	39.8	38.1	31.9	84.5	38.6	85.3	61.3	30.2	85.4	33.1	41.6	14.4	27.3	44.0	49.7
DT-ST (Ours)	✓	93.5	57.6	84.7	36.5	25.2	33.4	44.7	36.7	86.8	42.8	81.3	62.3	37.2	88.1	48.7	50.6	35.5	48.3	59.1	55.4

Table 1. Experimental results for GTA5 → Cityscapes. 'SF' represents whether the method is in source-free setting. † denotes the re-implementation in source-free setting.

	SF	road	sidewalk	Building	Wall	fence	pole	light	sign	vege.	sky	person	rider	car	bus	mbike	bike	mIoU	mIoU*
IAST (ECCV'20) [40]	✗	81.9	41.5	83.3	17.7	4.6	32.3	30.9	28.8	83.4	85.0	65.5	30.8	86.5	38.2	33.1	52.7	49.8	57.0
MetaCorr (CVPR'21) [14]	✗	92.6	52.7	81.3	8.9	2.4	28.1	13.0	7.3	83.5	85.0	60.1	19.7	84.8	37.2	21.5	43.9	45.1	52.5
ProDA (CVPR'2021) [59]	✗	87.1	44	83.2	26.9	0.7	42	45.8	34.2	86.7	81.3	68.4	22.1	87.7	50	31.4	38.6	51.9	58.5
SAC (CVPR'2021) [1]	✗	89.3	47.2	85.5	26.5	1.3	43	45.5	32	87.1	89.3	63.6	25.4	86.9	35.6	30.4	53	52.6	59.3
CPST (CVPR'2022) [31]	✗	87.3	44.4	83.8	25.0	0.4	42.9	47.5	32.4	86.5	83.3	69.6	29.1	89.4	52.1	42.6	54.1	54.4	61.7
Source model		52.2	23.6	62.2	6.0	0.2	28.3	7.3	12.7	79.7	75.7	52.5	10.2	75.0	24.6	8.9	10.3	33.1	38.1
URMDA (CVPR'2021) [9]	✓	59.3	24.6	77	14	1.8	31.5	18.3	32	83.1	80.4	46.3	17.8	76.7	17	18.5	34.6	39.6	45
SFDA (CVPR'2021) [57]	✓	67.8	31.9	77.1	8.3	1.1	35.9	21.2	26.7	79.8	79.4	58.8	27.3	80.4	25.3	19.5	37.4	42.4	48.7
SDF (MM'2021) [57]	✓	90.9	45.5	80.8	3.6	0.5	28.6	8.5	26.1	83.4	83.6	55.2	25	79.5	32.8	20.2	43.9	44.2	51.9
HCL (NIPS'2021) [21]	✓	80.9	34.9	76.7	6.6	0.2	36.1	20.1	28.2	79.1	83.1	55.6	25.6	78.8	32.7	24.1	32.7	43.5	50.2
DT-ST (Ours)	✓	79.4	41.4	73.9	5.9	1.5	30.6	35.3	19.8	86.0	86.0	63.8	28.6	86.3	36.6	35.2	53.2	47.7	55.8
Source model + DG [32]		76.8	29.8	67.9	10.7	0.3	29.5	9.5	16.8	79.8	78.3	52.5	13.8	78.5	28.5	12.8	19.9	37.8	43.5
ProDA†(CVPR'2021) [59]	✓	79.9	35.7	75.5	20.7	0	39.6	36.5	31.5	84.2	80.6	64.2	9.6	85.3	40.9	24.9	35.8	46.6	52.7
SAC†(CVPR'2021) [1]	✓	84.7	39.6	80.9	16.3	0.2	38.4	40.9	27.4	82.5	84.7	59.1	16.6	82.3	31	20.8	36.1	46.3	52.8
CPST†(CVPR'2022) [31]	✓	80.9	28.7	81	20.4	1.2	38.6	36.3	31.4	85.3	74.4	64.2	12.6	87.2	31.9	16.3	42.8	45.8	51.8
HCL (NIPS'2021) [21]	✓	86.7	38.1	82.7	10.0	0.6	30.3	25.4	29.7	82.8	85.9	61.9	24.8	84.5	38.9	22.6	37.9	46.4	54.0
DT-ST (Ours)	✓	88.9	45.8	83.3	13.7	0.8	32.7	31.6	20.8	85.7	82.5	64.4	27.8	88.1	50.9	37.6	57.3	50.7	58.8

Table 2. Experimental results for SYNTHIA → Cityscapes. Conforms to the same definition as in Table 1.

source-trained models on each MDA benchmark task, and in fact, achieves new state-of-the-art performance. Furthermore, our method achieves competitive performance compared to state-of-the-art UDA methods, despite not accessing source data during the adaptation process.

GTA5 → Cityscapes We report the experimental results of GTA5 → Cityscapes in Table 1. In comparison with UMA methods, using original source-trained models (*i.e.* 36.8% mIoU on validation set) for adaptation, our method achieved a 52.1% mIoU score, outperforms the state-of-the-art method SDF [57] by 2.7%. Note that, SDF performs multiple rounds of offline training to improve performance, while our method is performed online. Compared with online self-training HCL [21], our method achieves better adaptation performance, which exceeds the mIoU score of HCL by 4%. In that case, our method is comparable to the UDA methods IAST [40] and MetaCorr [14]. In addition, we also use the model trained by domain generalization technology [32] to perform adaptation, which can provide better pre-training. Benefit from this, our method further improved the mIoU score by 3.3%, and achieved the performance comparable to the existing SOTA UDA method CPST [31]. Besides, our method gains a 1.6% mIoU score benefit than HCL, which shows that our method can mine the knowledge in the model more efficiently.

SYNTHIA → Cityscapes We report the results of SYNTHIA → Cityscapes in Table 2. Due to the large domain shift in this task and the poor performance of the source-

Source GTA→	SF	Compound			Open	Avg	
		Rainy	Snowy	Cloudy	Overcast	C	C+O
Source Only		19.7	18.4	20.5	22.5	19.7	21.0
CBST [63]	✗	21.3	20.6	23.9	24.7	22.2	22.6
IBN-Net [42]	✗	20.6	21.9	26.1	25.5	22.8	23.5
PyCDA [34]	✗	21.7	22.3	25.9	25.4	23.3	23.8
OCDA [37]	✗	22.0	22.9	27.0	27.9	24.5	25.0
MOCDA [12]	✗	24.4	27.5	30.1	31.4	27.7	29.4
HCL [21]	✓	22.8	25.8	28.6	27.7	25.9	26.2
DT-ST (Ours)	✓	**26.7**	**28.1**	**32.1**	**32.5**	**30.1**	**31.3**

Table 3. Experimental results for GTA5 → BDD-100k using VGG-16 as backbone following [12]. 'SF' represents whether the method is in source-free setting.

source-trained model	Base ST	DTU-E	DTU-SND	TCR	mIoU	gain
original source					36.8	
	✓				46.2	+9.4
	✓	✓			47.2	+10.4
	✓		✓		47.8	+11.0
	✓	✓		✓	51.5	+14.7
	✓		✓	✓	52.3	+15.5
domain generation [32]					43.3	
	✓				50.7	+7.4
	✓	✓			51.3	+8.0
	✓		✓		52.6	+9.3
	✓	✓		✓	54.5	+11.2
	✓		✓	✓	55.4	+12.1

Table 4. Ablation study. We report mIoU scores (%) (val) using two source-trained models on GTA5 → Cityscapes adaptation task Under UMA setting.

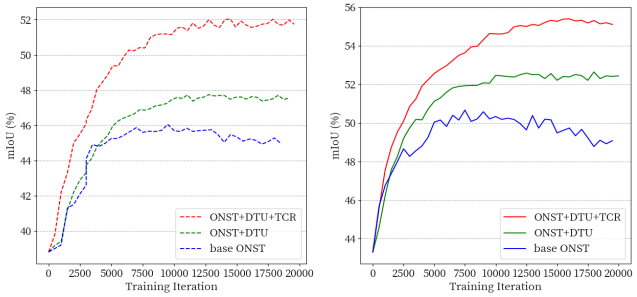

Figure 4. The validation mIoU score curve of adding DTU and TCR. The left adopts the original source model, and the right adopts the trained model by domain generalization.

trained model, most UMA methods obtain low adaptation benefits in this task. Nevertheless, our method still maintains good model adaptability. In UMA setting, using original source-trained model (*i.e.* 38.1% mIoU over 16 categories on validation set), our method achieves the mIoU score by 47.7% and 55.8% over the 16 and 13 categories separately. Compared with the source-trained model, our method improves the mIoU score by 17.1% and 14.6% over the 16 and 13 categories separately, which is better than the currently published performance of SDF [57] by 3.5% and 3.9%. In addition, using pre-training model by domain generalization method [32], our method further improves the mIoU score by 3.0% over the 16 and 13 categories, and achieves comparable performance with the UDA methods IAST [40], SAC [1], and ProDA [59].

GTA5 → BDD100k We carry out the MDA experiment of GTA5 → BDD100k to verify the proposed method on complex adaptation scenarios, *i.e.*, single domain to multiple target domains. The results are reported in Table 3. In this task, worse pre-train models and challenging adaptation scenarios are given, which further tests the stability and adaptability of the MDA methods. Under the same settings as MOCDA [12], our method achieved 30.1% average mIoU score in rainy, sunny, and cloudy scenarios, with an average increase of 10.4% mIoU score. In contrast, the UMA method HCL [21] does not achieve the desired adaptation effect. Compared with the state-of-the art UDA method MOCDA using meta-learning, our method has a 2.4% higher mIoU score, illustrating the potential of our method to adapt to complex scenarios and tasks.

4.3. Ablation study

In this part, we perform ablation experiments on GTA5 → Cityscapes task using two source-trained models, as shown in Table 4. The two source-trained methods achieve mIoU scores of 36.8% and 43.3%, respectively. The basic online self-training (ONST) provides a strong baseline, which improves the mIoU score by 9.4% and 7.4% using the two source-trained methods, respectively. On this basis, we add dynamic teacher update (DTU) mechanism and training consistency based resampling (TCR) strategy in turn. Both DTU-E and DTU-SND improve the performance of the base ONST, with DTU-SND being even better, suggesting that controlling the teacher update interval to incorporate an evolved student model can improve adaptability. In particular, DTU-SND improves the mIoU score of base ONST by 1.6% and 1.9%, respectively. Then TCR is added, and the performance is further improved on different basic models and strategies by more than 3% mIoU scores, which shows that TCR plays a crucial role in alleviating the imbalance problem in ONST.

To show training stability, we plot the mIoU score of the model on the validation set during the training phase, as shown in Fig. 4. The mIoU score of the base ONST (blue curve) increases steadily at the beginning but gradually decreases at the later stage, showing the poor stability of the base ONST. The mIoU score (green curve) with the added DTU-SND rises steadily at the beginning and then stabilizes, which stresses that DTU can effectively prevent ONST degradation. On this basis, the performance of adding TCR (red curve) shows faster improvement and more stability, which means that TCR can further stabilize training. Different source-trained models present similar training trends, further confirming the above conclusions.

4.4. Hyperparameter sensitivity

In this part, we analyze the sensitivity of hyperparameters in the dynamic teacher update (DTU) mechanism and training-consistency based resampling (TCR) strategy.

In DTU, we explore the effect of moving average weights γ in Eq. 3. When γ varies in the range of 0.99-0.999, the corresponding mIoU score and the distribution of the update intervals is shown in Fig. 5. It shows that changing γ within a certain range has little effect on the performance. Moreover, the update interval of the teacher can be adaptively adjusted according to γ. When γ is small (blue), the update degree of the teacher's weight is large each time. Correspondingly, the update interval tends to be larger, meaning that students take longer to evolve. This further presents the advantages of DTU than fixed interval.

In TCR, we explore the the impact of reliability evaluation frequency I and top $C\%$ candidates on performance, as shown in Table 5. Overall, the two parameters I and C have little effect on the adaptation performance, and the variance is within 1.4%. From the changing trend of mIoU score, we find that picking more candidates for copy-paste is benefit for adaptation. When C is selected above 50, the performance fluctuation is smaller with only 0.7% variance, showing that TCR is robust to both parameters.

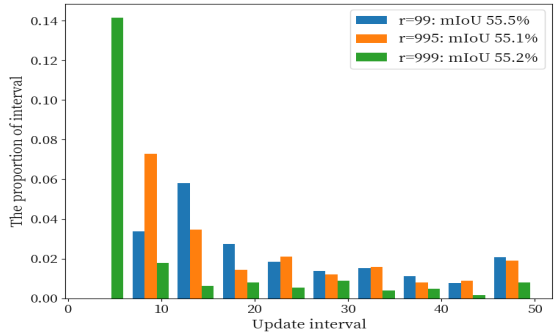

Figure 5. The distribution of the update intervals by DTU with varying γ.

$C\downarrow$	$I\rightarrow$	2000	3000	4000	5000
30		54.1	54.2	54.3	54.1
40		54.3	54.5	54.6	54.8
50		54.8	55.2	55.5	55.1
60		54.8	55.1	55.3	55.2

Table 5. The mIoU (%) score on GTA5 $\rightarrow$ Cityscapes (val) with varying $C\%$ and I using source-trained model by domain generalization

.

4.5. Qualitative assessment

To further demonstrate that our method promotes the evolution of teacher models in UMA, we visualize the output of teacher models with different iterations in GTA5 $\rightarrow$ Cityscapes task, as shown in Fig. 6. At the beginning of training, the output of the teacher model is chaotic and noisy. With training, the noise is weakened, and the prediction of the teacher model is gradually reasonable and regular. The above discussion further verifies that our method can steadily evolve the teacher model in UMA.

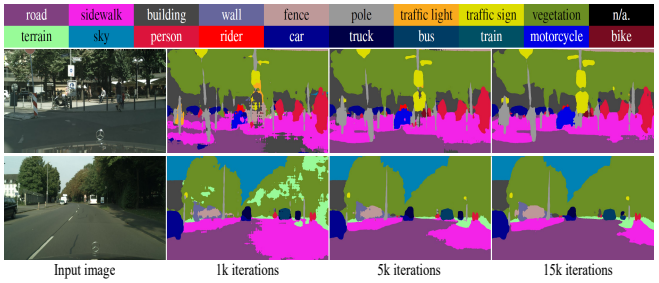

Figure 6. Visualization of output from the teacher model for different iterations. Our method can stabilize the teacher model and further achieve co-evolution in UMA.

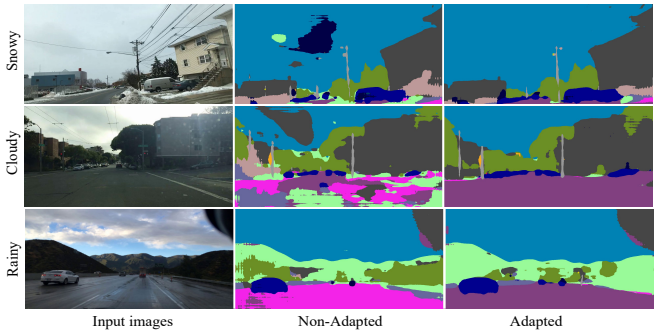

Figure 7. Visualization of the adaptation results of our method from single-domain to mixed-domain, including rainy, snowy and cloudy scenes (*i.e.* GTA5 $\rightarrow$ BDD-100K).

Fig. 7 shows the visualization of the results of our method performing UMA in a more challenging scenario, *i.e.* GTA5-BDD100k. It shows that our method can achieve good self-training performance in the face of difficult adaptation scenarios, even if only poor source-trained models are given.

5. Conclusion

In this paper, we focus on the problem of domain adaptive semantic segmentation when the source domain is inaccessible. We explore the reasons for the impaired stability and adaptability of online self-training, and propose corresponding improvement schemes. On multiple standard UMA benchmarks, our method greatly improves the stability and adaptability of online self-training methods, achieving comparable or even better performance than state-of-the-art UDA methods. We hope that our work can inspire the community and promote the performance of online self-training in more complex scenarios.

References

[1] Nikita Araslanov and Stefan Roth. Self-supervised augmentation consistency for adapting semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15384–15394, 2021. 1, 2, 3, 6, 7

[2] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010. 5

[3] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. 5

[4] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1791–1800, 2019. 2

[5] Yi-Hsin Chen, Wei-Yu Chen, Yu-Ting Chen, Bo-Cheng Tsai, Yu-Chiang Frank Wang, and Min Sun. No more discrimination: Cross city adaptation of road scene segmenters. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2017. 1

[6] Boris Chidlovskii, Stephane Clinchant, and Gabriela Csurka. Domain adaptation in the absence of source domain data. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 451–460, 2016. 1

[7] Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6830–6840, 2019. 2

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5

[9] Francois Fleuret et al. Uncertainty reduction for model adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9613–9623, 2021. 1, 2, 6

[10] Li Gao, Jing Zhang, Lefei Zhang, and Dacheng Tao. Dsp: Dual soft-paste for unsupervised domain adaptive semantic segmentation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2825–2833, 2021. 2, 3, 5

[11] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2918–2928, 2021. 2, 3, 5

[12] Rui Gong, Yuhua Chen, Danda Pani Paudel, Yawei Li, Ajad Chhatkuli, Wen Li, Dengxin Dai, and Luc Van Gool. Cluster, split, fuse, and update: Meta-learning for open compound domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8344–8354, 2021. 1, 5, 7

[13] Yves Grandvalet, Yoshua Bengio, et al. Semi-supervised learning by entropy minimization. *CAP*, 367:281–296, 2005. 2, 4

[14] Xiaoqing Guo, Chen Yang, Baopu Li, and Yixuan Yuan. Metacorrection: Domain-aware meta loss correction for unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3927–3936, June 2021. 2, 6

[15] Jianzhong He, Xu Jia, Shuaijun Chen, and Jianzhuang Liu. Multi-source domain adaptation with collaborative learning for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11008–11017, 2021. 1

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5

[17] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. CyCADA: Cycle-consistent adversarial domain adaptation. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1989–1998, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. 2

[18] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *CoRR*, abs/1612.02649, 2016. 1

[19] Weixiang Hong, Zhenzhen Wang, Ming Yang, and Junsong Yuan. Conditional generative adversarial network for structured domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[20] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. DAFormer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9924–9935, 2022. 2, 3

[21] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. *Advances in Neural Information Processing Systems*, 34:3635–3649, 2021. 1, 2, 3, 5, 6, 7

[22] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Rda: Robust domain adaptation via fourier adversarial attacking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8988–8999, 2021. 2

[23] Jiaxing Huang, Shijian Lu, Dayan Guan, and Xiaobing Zhang. Contextual-relation consistent domain adaptation for semantic segmentation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 705–722, 2020. 2

[24] Guoliang Kang, Yunchao Wei, Yi Yang, Yueting Zhuang, and Alexander Hauptmann. Pixel-level cycle association: A new perspective for domain adaptive semantic segmentation. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3569–3580. Curran Associates, Inc., 2020. 2

[25] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[26] Jogendra Nath Kundu, Akshay Kulkarni, Amit Singh, Varun Jampani, and R Venkatesh Babu. Generalize then adapt: Source-free domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7046–7056, 2021. 1, 2

[27] Jogendra Nath Kundu, Akshay R Kulkarni, Suvaansh Bhambri, Deepesh Mehta, Shreyas Anand Kulkarni, Varun Jampani, and Venkatesh Babu Radhakrishnan. Balancing discriminability and transferability for source-free domain adaptation. In *International Conference on Machine Learning*, pages 11710–11728. PMLR, 2022. 1, 2

[28] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. Content-consistent matching for domain adaptive semantic segmentation. In *ECCV*, pages 440–456. Springer, 2020. 2

[29] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. Content-consistent matching for domain adaptive semantic segmentation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 440–456, 2020. 2

[30] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9641–9650, 2020. 1

[31] Ruihuang Li, Shuai Li, Chenhang He, Yabin Zhang, Xu Jia, and Lei Zhang. Class-balanced pixel-level self-labeling for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11593–11603, 2022. 1, 2, 6, 7

[32] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and LINGYU DUAN. Uncertainty modeling for out-of-distribution generalization. In *International Conference on Learning Representations*, 2022. 6, 7

[33] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[34] Qing Lian, Fengmao Lv, Lixin Duan, and Boqing Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6758–6767, 2019. 2, 7

[35] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020. 1

[36] Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1215–1224, 2021. 1, 5, 6

[37] Ziwei Liu, Zhongqi Miao, Xingang Pan, Xiaohang Zhan, Dahua Lin, Stella X Yu, and Boqing Gong. Open compound domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12406–12415, 2020. 7

[38] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Significance-aware information bottleneck for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[39] Yawei Luo, L. Zheng, T. Guan, Junqing Yu, and Y. Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2502–2511, 2019. 2

[40] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. 2020. 2, 6, 7

[41] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[42] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 464–479, 2018. 7

[43] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 102–118. Springer International Publishing, 2016. 5

[44] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5

[45] Kuniaki Saito, Donghyun Kim, Piotr Teterwak, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Tune it the right way: Unsupervised validation of domain adaptation via soft neighborhood density. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9184–9193, 2021. 2, 4

[46] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3752–3761, 2018. 1

[47] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[48] M.Naseer Subhani and Mohsen Ali. Learning from scale-invariant examples for domain adaptation in semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[49] Marco Toldo, Andrea Maracani, Umberto Michieli, and Pietro Zanuttigh. Unsupervised domain adaptation in semantic segmentation: a review. *Technologies*, 8(2):35, 2020. 1

[50] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1379–1389, 2021. 2, 3, 5

[51] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[52] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Perez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[53] Haoran Wang, Tong Shen, Wei Zhang, Ling-Yu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 642–659, 2020. 2

[54] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogerio Feris, Jinjun Xiong, Wen-mei Hwu, Thomas S. Huang, and Honghui Shi. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[55] Haifeng Xia, Handong Zhao, and Zhengming Ding. Adaptive adversarial network for source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9010–9019, 2021. 1, 2

[56] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[57] Mucong Ye, Jing Zhang, Jinpeng Ouyang, and Ding Yuan. Source data-free unsupervised domain adaptation for semantic segmentation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2233–2242, 2021. 1, 2, 6, 7

[58] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020. 5

[59] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12414–12424, 2021. 1, 2, 3, 5, 6, 7

[60] Y. Zhang, P. David, H. Foroosh, and B. Gong. A curriculum domain adaptation approach to the semantic segmentation of urban scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(8):1823–1841, 2020. 1

[61] Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei. Fully convolutional adaptation networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2

[62] Xinge Zhu, Hui Zhou, Ceyuan Yang, Jianping Shi, and Dahua Lin. Penalizing top performers: Conservative loss for semantic segmentation adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 568–583, 2018. 2

[63] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018. 1, 2, 7

[64] Yang Zou, Zhiding Yu, Xiaofeng Liu, B. V. Kumar, and J. Wang. Confidence regularized self-training. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5981–5990, 2019. 2